AF/ZKW

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

John M. GARTH, et al.

Appln. No.: 10/082,327

Confirmation No.: 3782

Filed: February 26, 2002

Docket No: A8508

Group Art Unit: 2167

Examiner: Susan F. Rayyan

For: SYSTEM AND METHOD FOR PREDICTING EXECUTION TIME OF A DATABASE UTILITY COMMAND

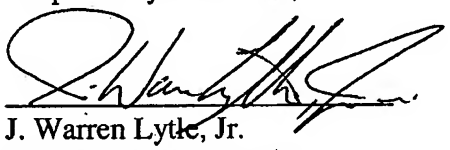
SUBMISSION OF APPEAL BRIEF

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith please find an Appeal Brief. The USPTO is directed and authorized to charge the statutory fee of $500.00 and all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account. A duplicate copy of this paper is attached.

Respectfully submitted,



J. Warren Lytle, Jr.
Registration No. 39,283

SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE
23373
CUSTOMER NUMBER

Date: September 21, 2005

PATENT APPLICATION

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of

John M. GARTH, et al.

Appln. No.: 10/082,327

Confirmation No.: 3782

Filed: February 26, 2002

Docket No: A8508

Group Art Unit: 2167

Examiner: Susan F. Rayyan

For: SYSTEM AND METHOD FOR PREDICTING EXECUTION TIME OF A DATABASE UTILITY COMMAND

APPEAL BRIEF UNDER 37 C.F.R. § 41.37

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In accordance with the provisions of 37 C.F.R. § 41.37, Appellant submits the following:

Table of Contents

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines Corporation of Armonk, New York, the assignee of the present application. The assignment was recorded on February 26, 2002 at Reel 012643, Frame 0402.

II. RELATED APPEALS AND INTERFERENCES

To the best of the knowledge and belief of the Appellant, the Assignee and the undersigned, there are no other appeals or interferences before the Board of Appeals and Interferences ("the Board") that will directly affect, or be affected by, the Board's decision in the present Appeal.

III. STATUS OF CLAIMS

The following is a statement of the status of all claims in the proceeding and an identification of those claims that are being appealed.

Claims 1-36 are all the claims pending in the present application, each of which stands rejected as follows.

Claims 1, 2, 17-30 and 33-36 are rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 5,121,494 to Dias et al. ("Dias") and U.S. Patent No. 6,469,751 to Isobe et al. ("Isobe").

Claims 3-9 and 12-14 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Dias and Isobe in view of the article entitled "Method of Sharing an Intelligent Progress Bar Across Remote Machines, IBM, 1994 ("Sharing").

Claim 32 is rejected under 35 U.S.C. § 103(a) as being unpatentable over Dias and Isobe in view of Sharing, and further in view of U.S. Published Patent Application No. 2003/0131146 to Lam et al. ("Lam").

The rejection of each of these pending claims is being appealed.

A copy of the claims on appeal is set forth in an attached Appendix.

IV. STATUS OF AMENDMENTS

A Response under 37 C.F.R. § 1.116 was filed on June 13, 2005, in response to the Final Office Action dated March 21, 2005. In an Advisory Action dated July 18, 2005, the Examiner states that the Response filed June 13, 2005, has been considered but did not place the application in a condition for allowance. No changes were made to the claim set by way of the June 16, 2005 Response, and no other amendment or response was filed subsequent to the April 20, 2005 Final Office Action.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

The following is a concise explanation of the subject matter defined in each of the independent claims involved in the appeal.

For the Board's convenience, Appellants will first describe the relevant art (pp. 1-3 of the Specification), and then each of the independent claims with reference to the exemplary embodiments of the invention (pp. 5-20 of the Specification). This discussion of the exemplary embodiments and the pending claims is provided for explanatory purposes only, and is not intended to limit the scope of the claims.

Generally, the invention relates to computer database utilities for predicting the time required to execute database utility commands. See pg. 1, line 4.

The Related Art

A problem facing database administrators is that they often must execute database utility commands during limited periods of time when the utilities will cause the least amount of disruption as possible to database uses. See pg. 1, lines 6-18. These database administrators want to know how long each utility command will take so the administrator can schedule the utility commands with some confidence that they will complete in the batch window of time set aside for running utility commands. See pg. 2, lines 2-4.

A conventional approach for estimating the amount of time that a particular utility command will take to execute is to break down the utility command into its constituent sub-utility commands (such as reading or writing database records), apply rule-of-thumb formulas based upon the average time needed to complete such sub-utility commands, and compute a time

estimate based upon the rule-of-thumb formulas tailored with relevant information about the database. See pg. 2, lines 5-10. However, the conventional approach suffers from at least two problems. The first is that it is very labor intensive to supply the required parameters to the rule-of-thumb formula. A second problem is that these rule-of-thumb estimates might not be accurate enough to ensure the utility commands complete in required time. See pg. 2, lines 14-22. Accordingly, more precise methods and techniques are needed for predicting the time required to execute utility commands on a database. See pg. 2, lines 7-11.

Claim 1

Claim 1 is directed to a method for predicting the time required to execute a database command. The claim recites measuring a plurality of execution times to complete the database command, and recording the measured execution times, thereby creating a time historical record. An embodiment of these steps is shown in Fig. 2 at steps 250 and 260 and in the steps of Fig. 5. See also pg. 8, line 7 through pg. 10, line 14 and pg. 16, line 1 through pg. 17, line 12.

The claim also recites using the historical time record to estimate the time required to execute the database command. An embodiment of this step is shown in Fig. 3 and described at pg. 5, line 11 through pg. 6, line 14 and pg. 10, line 15 through pg. 15, line 3.

Claim 28

Claim 28 is directed to an apparatus for predicting the time required to execute a database command. This claim recites a historical record module having recorded therein a plurality of

measurements of execution times of the database command, which is shown, for example, by the database management system 110 in Fig. 1. See also, pg. 7, lines 20-22.

An example of an analysis module configured to analyze the measurements recorded in the historical record module, is shown by the statistical analysis unit 108 shown in Fig. 1. See also, pg. 7, lines 20 through pg. 8, line 2.

An example of the utility scheduling module configured to determine whether to execute the database command based on an analysis of the database command measurements is shown, for example, as the utility scheduling tool 106 in Fig. 1 executing the steps shown in Fig. 2, which is described at pg. 9, line 1 through pg. 10, line 7, and in Figs. 3 and 4.

Claim 33

Claim 33 is directed to an apparatus for predicting a time for executing a database command. The apparatus includes means for measuring a time to complete a database command. Structure for performing that function is disclosed, for example, at least by a computer executing step 250 in Fig. 2 and at pg. 20, lines 3-6.

The apparatus also includes means for recording the measured time, thereby creating a historical time record. Structure for performing that function is disclosed, for example, at least by a computer executing step 260 in Fig. 2 and at pg. 20, lines 3-6.

The apparatus further includes means for analyzing the time historical record to estimate the time required to execute the database utility command. Structure for performing that function is disclosed, for example, at least by a computer executing step 220 in Fig. 2 and at pg. 20, lines 3-6.

Claim 35

Claim 35 is directed to a computer program embodied on a computer readable medium for predicting a time for executing a database command. See pg. 19, line 17 through pg. 20, line 6. The claim recites program instructions for measuring a time to complete execution of the database command. The claim also recites program instructions for recording the measured time, thereby creating a time historical record. An embodiment of these steps, performed by executing program instructions, is shown in Fig. 2 at steps 250 and 260 and in the steps of Fig. 5. See also pg. 8, line 7 through pg. 10, line 14 and pg. 16, line 1 through pg. 17, line 12.

The claim further recites program instructions for analyzing the time historical record to estimate the time required to execute the database utility command. An example of analyzing the measurements recorded in the historical record module is shown by the statistical analysis step 220 shown in Fig. 2 and in Figs. 3 and 4. See also, pg. 8, line 22 through pg. 9, line 19 and pg. 10, line 15 through pg. 11, line 5, and pg. 15, line 4 through pg. 16, line 11.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The following is a concise statement of each ground of rejection presented for review.

1. Whether claims 1, 2, 17-30 and 33-36 are rendered unpatentable under 35 U.S.C. § 103(a) over Dias and Isobe.

2. Whether claims 3-9 and 12-14 are rendered unpatentable under 35 U.S.C. § 103(a) over Dias and Isobe in view of Sharing.

3. Whether claim 32 is rendered unpatentable under 35 U.S.C. § 103(a) over Dias and Isobe in view of Sharing, and further in view of Lam:

VII. ARGUMENT

Rejection of Claims 1, 2, 17, 30 and 33-36

Claims 1, 2, 17-30 and 33-36 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Dias and Isobe. Appellant submits that the Dias/Isobe combination set forth in the final Office Action does not render these claims unpatentable because the Examiner has not established a *prima facie* case of obviousness and because a person of ordinary skill in the art would not have been motivated to combine the references as the Examiner asserts.

The Prior Art References

A brief discussion of the references the Examiner cites in support of the claim rejections is presented here for the Board's convenience.

1. Dias

Dias is directed to a technique for performing a database join operation in a multiple processor database system in which the technique effectively deals with a problem known as data skew. See Abstract.

Dias' technique performs a join operation in three stages. Col. 2, lines 55-57. The first stage is a preprocessing stage the results of which are used in a second stage that defines subtasks for the final join operation. The second stage then allocates those subtasks to different processors in such a manner that the processors are close to equally loaded in the final join operation. Col. 2, lines 55-57 and 22-68. The second stage will further partition the subtasks to balance sort operations among the several processors. Col. 3, lines 25-29. Dias describes this second stage as being "crucial to the invention" (col. 5, lines 1-2) as the purpose of the second

stage is to "optimally allocate these subtasks to different processors...." Col. 4, lines 64-66. The third stage performs the actual join operation using the results of the second stage. Col. 3, lines 4-12.

Dias discloses, for example, at col. 3, lines 26-30, that the second stage estimates the time it will take to perform each of the subtasks to balance sort operations among the processors. Jobs are scheduled among the processors according to optimization techniques and based on estimated skew. Col. 3, lines 29-34.

Dias creates the subtasks based on the number of rows to be processed by a subtask. See col. 5, lines 45-60, reproduced below.

> will become apparent in the following description. Let us estimate the time it takes to perform this subtask as $T^1_{v_1,v_2} = A(I_1 + I_2) + BO$, where
>
> $$I_i = \sum_{j=1}^{P} \text{card}(\rho_{ij}, v_1, v_2)$$
>
> is the number of rows of input from R_i, O is the number of rows of output from the merge of R_1 and R_2, and A and B are known constants. If we make the assumption

As Dias discloses, the estimated time for a subtask to execute, $T^1v_1v_2$, is a function of I_1 and I_2 which are the "number of rows of input from R_i" and O which is "the number of rows of output from a merge of R_1 and R_2." Dias' goal is to create subtasks that each execute for approximately the same amount of time on the several processors so as to minimize the total execution time. See col. 6, lines 45 through col. 7, line 1. To accomplish this goal, Dias dynamically defines the subtasks based on the number of rows to be processed so that the execution times of the subtasks are approximately equal. See generally, col. 7, lines 14-20 and

Fig. 4 which shows an algorithm for dynamically creating subtasks. As shown in step 32, if a subtask is estimated to take too long to process, based on the number or rows to process, the subtask is split so that it processes fewer rows. Col. 7, lines 43-62. In this way, based on the number or rows to process, Dias dynamically optimizes execution of a database join operation.

2. Isobe

Isobe relates to a remote control device for controlling various types of devices, such as a television set 102 (see Fig. 1), a video set 103, a lamp 101, and an audio-visual amplifier 104. Col. 7, lines 1-10.

Isobe's remoter controller 300, shown in Fig. 1, estimates a series or string of remote controller commands that a user inputs based on the frequency of occurrence of the command string or the time of day the command string has been used in the past. Col. 3, lines 1-10 and 20-28. Isobe's remote controller includes a database of command sequences, as shown in Fig. 14A. These command sequences are sequences of remote controller functions, represented by codes such as A1, A2, or B3. A sequence of these codes, such as A1A2 corresponds to a remote control function, such as changing the channel on a television. Col. 3, lines 30-35 and col. 8, lines 24-45.

Isobe, beginning at col. 19, line 9, describes a command estimating process, shown in Fig. 13, for estimating a command sequence based on a user's operation of a remote control device. Isobe shows, in Fig. 14C, a compilation of command strings and a frequency of occurrence of each such string. As a command is input (e.g., A1) a probability for each string that begins with that command in computed, as shown in Fig. 14D. If the probability exceeds a

threshold, such as 55%, the estimator determines the command string to output. In Fig. 14D the string A1A2A3 has a probability of 56% when the commands A1 and A2 are received. Thus, the estimator estimates the command string to be A1A2A3. Col. 19, line 53 through col. 20, line 42.

Isobe also discloses estimating a command string based on the day and time the command is received. Figs. 15A and 15B show a database of commands and times. Isobe describes, at col. 20, lines 55-59, another embodiment of a database in Fig. 16 indicating the number of occurrences of a command in predetermined time intervals (e.g., Monday at 20:00 +/- 10 minutes) according to day and time. The estimating unit determines a command's frequency of usage at the time command inputs are received. If the frequency of occurrence for a command string exceeds a threshold based on the day and time information in the database of Fig. 16, that command string is output. See cols. 20 and 21.

3. Sharing

Sharing relates to a progress bar (PB) of an end user interface that informs a user of the progress of an application program's execution. Sharing's progress bar "teaches itself how to predict the amount of time particular applications will need, based on previous and similar executions of that application which are contained in the statistical data base." Sharing, second paragraph. Upon completion of the application's execution "the system records the actual execution time in the data base for subsequent use." Sharing, third paragraph. The progress bar provides feedback to a user as to the progress of the applications execution, such as providing an audible update of "10% complete" and "100% complete." Sharing, first paragraph of second page of reference.

Sharing also discloses "the ability to share information learned by the PB with other machines ..." Sharing, fourth paragraph. When the progress bar does not have any historical information about the execution time of an application (Application A) on a particular machine, Dias discloses that "the system checks the PB tables on all other machines on the LAN, the data for Application A is averaged, and made available to Requestor A." Sharing, fourth paragraph.

4. Lam

Lam relates to controlling and monitoring computer command execution within a computer network or cluster. Para. [0002]. Lam describes the use of computer command scripts to configure computers in a network. Para. [0004]. More specifically, Lam discloses monitoring and control of computer script commands contained within a command file. Para. [0010]. This monitoring is performed by the computer via a graphical user interface (GUI) that allows the user to select a command for execution. *Id.*

The Rejection of the Independent Claims

Claim 1, for example, is directed to a method for predicting the time required to execute a database command. The method includes i) measuring a plurality of execution times to complete the database command; ii) recording the measured execution times, thereby creating a time historical record; and iii) using the time historical record to estimate the time required to execute the database command.

The Examiner asserts that Dias, at col. 3, lines 26-27 and col. 8, lines 47-53, teaches measuring a plurality of execution times to complete a database command. The Examiner

admits that Dias does not teach recording the measured execution times and using the time historical record to estimate the time required to execute the database command. To overcome that deficiency, the Examiner cites Isobe. Specifically, the Examiner cites the log shown in Fig. 15A and discussed at col. 20, lines 59-62 of Isobe, as teaching these limitations.

The Examiner Fails to Establish a Prima Facie case of Obviousness Because the Dias/Isobe Combination Does Not Meet All the Limitations in the Claims.

Claim 1 recites "measuring a plurality of execution times to complete the database command; recording the measured execution times, thereby creating a time historical record; and using the time historical record to estimate the time required to execute the database command."

In the Office Action, the Examiner relies upon Dias for teaching measuring a plurality of execution times to complete a database command. The Examiner refers to the subtasks disclosed in Dias at col. 3, lines 26-27 and col. 8, lines 47-53, that are allocated to different processors to perform a database join operation, as the claimed database command. *See* Dias col. 2, lines 59-67. Although Dias discloses at col. 3, lines 26-27, estimating the time it would take for a single processor to perform each of the subtasks in order to balance operations among several processors, Dias does not disclose, or even suggest, measuring a plurality of execution times to complete a database command as required by claim 1. Rather, Dias estimates the time for a subtask to execute based on the number of database rows that will be operated on by the subtask, not by measuring a plurality of execution times to complete the subtask. Col. 5, lines 47-56 ("Let us estimate the time it takes to perform this subtask as $T^1_{v1v2}.= A(I_1+I_2)+BO$"). The estimate of the time it takes to execute a subtask, T^1_{v1v2}, is based on I_1 and I_2, which are the number of rows of input from relations R_1 and R_2, respectively, and O, which is the number of rows of output

from the merge of R_1 and R_2. *Id.* Contrary to the Examiner's assertion, Dias does not disclose, or even suggest, measuring execution times of database commands as required by claim 1.

The Examiner relies on Isobe for creating a historical time record. However, even if Dias was modified by the teachings of Isobe as the Examiner asserts in the final Office Action, all the limitations of the claims are not met since the combination would not measure a plurality of execution times of database commands. Hence, the Examiner has failed to establish a *prima facie* case of obviousness with respect to claim 1 and the claims that depend from claim 1.

Independent Claims 28, 33 and 35

With respect to claim 28, the Dias/Isobe combination does not have a historical record module having recorded therein a plurality of measurements of execution times of a database command. As discussed above, neither Dias nor Isobe, alone or in combination, teach measuring a plurality of execution times of a database command. Accordingly, the Dias/Isobe combination does not render claim 28 unpatentable.

With respect to claim 33, the Dias/Isobe combination does not perform the function of measuring a time to complete a database command, since neither Dias nor Isobe, alone or in combination, teaches measuring the execution time of a database command, as discussed above. Hence, the Dias/Isobe combination does not render claim 33 unpatentable.

With respect to claim 35, the Dias/Isobe reference does not have program instructions for measuring the time to complete execution of a database command for the reasons discussed

above with respect to claim 33. Accordingly, the Dias/Isobe combination does not render claim 35 unpatentable.

Thus, the Examiner has failed to establish a *prima facie* case of obviousness with respect to each of the independent claims (i.e., claims 1, 28, 33 and 35) and the claims that depend therefrom, respectively.

There is No Motivation in the Prior Art to Combine the Teachings of the References as Asserted by the Examiner

According to the Office Action it would have been obvious to combine these portions of Dias and Isobe "to issue a command according to the issue schedule based on the database," citing Isobe col. 5, lines 16-18.

It is respectfully submitted that a person of ordinary skill in the art would not have been motivated to modify the teachings of Dias with the alleged historical time record of Isobe, since to make such a modification would alter the theory of operation of Dias. Isobe's record of database commands and the day and time they execute have nothing to do with Dias' decomposing a database join operation into subtasks and allocating those subtasks to a plurality of processors. To link the execution of Dias' subtasks to a particular day and time would completely change the fundamental operation of Dias and destroy its purpose of balancing execution subtasks, since that balancing is accomplished based on the number or rows to process not on a particular date and time the subtask is executed. Accordingly, it would not have been obvious to combine the teachings of the references as the Examiner asserts.

Rejection of Dependent Claims 3-9 and 12-14

Claims 3-9 and 12-14 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Dias and Isobe in view of Sharing. The Examiner relies on Sharing for disclosing averaging of historical time records of an application's execution time. Applicant respectfully traverses the rejection since these claims depend from claim 1 and Sharing does not satisfy the deficiencies of the Dias/Isobe combination.

Sharing relates to a progress bar of an end user interface that indicates the progress of an application program's execution in which average execution times for the application program can be used. Even if the Dias/Isobe combination were modified to use average execution times as disclosed by Sharing, the resultant combination would not satisfy all the limitations of the independent claims. As discussed above with respect to claim 1, the Dias/Isobe combination would not measure execution times of a database command. Rather, Dias uses the number of rows to be processed as an indication of an execution time of a subtask. Even if the Dias/Isobe combination were modified based on Sharing to use an average of the rows to be processed in estimating the execution time of a subtask, the limitations of the independent claims are not satisfied because the combination would not measure the execution times of subtasks. Accordingly, the Dias/Isobe/Sharing combination does not meet all the limitations of the claims, and hence, does not render claims 3-9 and 12-14 unpatentable.

Rejection of Dependent Claim 32

Claims 32 is rejected under 35 U.S.C. § 103(a) as being unpatentable over Dias, Isobe and Sharing, in view of Lam.

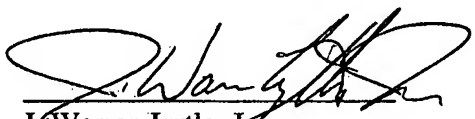
The Examiner relies on Lam for disclosing, at paragraph [0010], lines 6-9, a user interface module configured for enabling a user to specify the database command to be analyzed. Lam is directed to a graphical user interface (GUI) system for monitoring and controlling command scripts. See Abstract. The portion of Lam cited by the Examiner merely states that "[t]he plurality of computer commands are then displayed to the operator in a GUI display that allows the operator to accept a selection of at least one of the commands contained within the plurality of computer commands."

Appellant respectfully submits that even if the Dias/Isobe/Sharing combination were modified to use the GUI display disclosed by Lam, all the limitations of independent claim 28 would not be present, for the reasons discussed above regarding claim 28. That is, Lam does not satisfy the deficiencies of the Dias/Isobe combination as discussed above. Accordingly, the Dias/Isobe/Sharing/Lam combination does not render claim 32 unpatentable.

Unless a check is submitted herewith for the fee required under 37 C.F.R. §41.37(a) and 1.17(c), please charge said fee to Deposit Account No. 19-4880.

The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,



SUGHRUE MION, PLLC
Telephone: (202) 293-7060
Facsimile: (202) 293-7860

WASHINGTON OFFICE
23373
CUSTOMER NUMBER

J. Warren Lytle, Jr.
Registration No. 39,283

Date: September 21, 2005

CLAIMS APPENDIX

CLAIMS 1-36 ON APPEAL:

1. A method for predicting the time required to execute a database command, comprising:

measuring a plurality of execution times to complete the database command;

recording the measured execution times, thereby creating a time historical record; and

using the time historical record to estimate the time required to execute the database command.

2. The method of claim 1, wherein said using the time historical record includes analyzing the time historical record by using a statistical analysis technique to estimate the time required to execute the database command.

3. The method of claim 2, wherein said analyzing the time historical record includes computing an average execution time based upon information concerning the database command from the time historical record.

4. The method of claim 3, wherein the average execution time is $AvT = \sum M(i) / N$, where i is an integer and varies from 1 to N, N equals the number of measurements recorded in the historical record of the execution time of the database command, and M(i) is an ith measurement of the execution time of the database command.

5. The method of claim 2, wherein said analyzing the time historical record includes computing a moving range between prior measurements of the database command, based upon information from the time historical record.

6. The method of claim 5, wherein the moving range is MR(i)=M(i+1)-M(i), where i is an integer that varies from 1 to N, and M is a measurement of an execution time of the database command.

7. The method of claim 2, wherein said analyzing the time historical record includes computing a maximum execution time.

8. The method of claim 7, wherein said analyzing the time historical record includes computing the maximum execution time based upon a specified confidence value.

9. The method of claim 8, wherein the specified confidence value is configurable based upon a probability that the database command will execute in less time than the computed maximum execution time.

10. The method of claim 7, further comprising:

executing the database command;

measuring a time to execute the database command; and

issuing a warning if the measured time to execute the database command exceeds the maximum execution time.

11. The method of claim 10, wherein the warning is a warning that a configuration of the database may have changed.

12. The method of claim 2, wherein said analyzing the time historical record includes computing a minimum execution time.

13. The method of claim 12, wherein said analyzing the time historical record includes computing the minimum execution time based upon a specified confidence value.

14. The method of claim 13, wherein the specified confidence value is configurable based upon a probability that the database command will execute in less time than the computed minimum execution time.

15. The method of claim 12, further comprising:

executing the database command;

measuring a time to execute the database command; and

issuing a warning if the measured time to execute the database command is less than the minimum execution time.

16. The method of claim 15, wherein the warning is a warning that a configuration of the database may have changed.

17. The method of claim 1, wherein said database command is a database utility command.

18. The method of claim 17, further comprising recording within the time historical record the time of execution of said measured database utility command.

19. The method of claim 18, further comprising recording within the time historical record the day of execution of said measured database utility command.

20. The method of claim 17, further comprising recording within the time historical record a database utility command option executed with said measured database utility command.

21. The method of claim 17, further comprising recording within the time historical record a processor load of a computer executing said measured database utility command.

22. The method of claim 17, further comprising recording within the time historical record a storage access load of a computer executing said measured database utility command.

23. The method of claim 17, wherein using the time historical record further comprises selecting a historical record for analysis based upon one or more of the following:

a) the database utility command;

b) an option specified with a previously executed instance of the database utility command;

c) the time that the previously executed instance of the database utility command was executed;

d) the day that the previously executed instance of the database utility command was executed;

e) a processor load on a machine executing a previously executed instance of the database utility command; and

f) a storage access load on a machine executing a previously executed instance of the database utility command.

24. The method of claim 1, further comprising determining if a plurality of database commands can execute within a fixed timeframe by analyzing each of the plurality of commands based on prior execution time measurements for each of the plurality of database commands.

25. The method of claim 1, wherein said database command is a command for which a time required to execute has been estimated, the method further comprising:

editing the database command;

analyzing the time historical record using a statistical analysis technique to generate an estimate of the time required to execute the edited database command.

26. The method of claim 1, wherein said database command is a command file containing a plurality of database commands.

27. The method of claim 26, wherein times required to execute database commands within said command file have been estimated, the method further comprising:

editing the database command file;

analyzing the time historical record using a statistical analysis technique to generate new estimates of the time required to execute database commands contained within the edited command file.

28. An apparatus for predicting the time required to execute a database command, comprising:

a historical record module having recorded therein a plurality of measurements of execution times of the database command;

an analysis module coupled to the historical record module and configured to analyze the measurements recorded in the historical record module; and

a utility scheduling module configured to determine whether to execute the database command based on an analysis of the database command measurements.

29. The apparatus of claim 28, wherein the analysis module is configured to statistically analyze the time historical record to estimate the time required to execute the database command.

30. The apparatus of claim 28, wherein the database command is a database utility command.

31. The apparatus of claim 30, wherein the utility scheduling module is configured to determine whether a plurality of database commands can execute within a fixed timeframe based on the analysis module analyzing measurements relating the plurality of measurements recorded in the historical record module.

32. The apparatus of claim 30, further comprising a user interface module configured for enabling a user to specify the database command to be analyzed.

33. An apparatus for predicting a time for executing a database command, comprising:

means for measuring a time to complete the database command;

means for recording the measured time, thereby creating a time historical record; and

means for analyzing the time historical record to estimate the time required to execute the database utility command.

34. The apparatus of claim 33, wherein said means for analyzing uses a statistical analysis technique to analyze the time historical record to estimate the time required to execute the database command.

35. A computer program embodied on a computer readable medium for predicting a time for executing a database command, comprising:

program instructions for measuring a time to complete execution of the database command;

program instructions for recording the measured time, thereby creating a time historical record; and

program instructions for analyzing the time historical record to estimate the time required to execute the database utility command.

36. The computer program of claim 35, wherein said program instructions for analyzing use a statistical analysis technique to analyze the time historical record to estimate the time required to execute the database command.

EVIDENCE APPENDIX:

Pursuant to 37 C.F.R. § 41.37(c)(1)(ix), submitted herewith are copies of any evidence submitted pursuant to 37 C.F.R. §§ 1.130, 1.131, or 1.132 or any other evidence entered by the Examiner and relied upon by Appellant in the appeal.

Appellant submits that no such evidence has been submitted in this application on appeal.

RELATED PROCEEDINGS APPENDIX

Submitted herewith are copies of decisions rendered by a court or the Board in any proceeding identified about in Section II pursuant to 37 C.F.R. § 41.37(c)(1)(ii).

Appellant respectfully submits that no other proceedings have been identified in Section II and hence, no copies of any decisions rendered by a court or the Board are submitted herewith.